2013-1165

United States Court of Appeals for the Federal Circuit

NAZOMI COMMUNICATIONS, INC.,

Plaintiff-Appellant,

v.

NOKIA CORPORATION AND NOKIA INC.,

Defendants,

and

AMAZON.COM, INC.,

Defendant,

and

WESTERN DIGITAL CORPORATION AND
WESTERN DIGITAL TECHNOLOGIES, INC.,

Defendants-Appellees,

and

SLING MEDIA, INC.,

Defendant-Appellee,

and

VIZIO, INC.,

Defendant.

On Appeal from the United States District Court for the Northern District of California in Case No. 10-CV-4686, Senior Judge Ronald M. Whyte.

BRIEF FOR DEFENDANTS-APPELLEES

KEVIN P. ANDERSON
KARIN A. HESSLER
ROBERT J. SCHEFFEL
WILEY REIN LLP
1776 K Street, N.W.
Washington, D.C. 20006-2304

Counsel for Defendants-Appellees Western Digital Corporation and Western Digital Technologies, Inc.

DEANNE E. MAYNARD
BRIAN R. MATSUI
MORRISON & FOERSTER LLP
2000 Pennsylvania Avenue, N.W.
Washington, D.C. 20006-1888
Telephone: 202.887.8740

RACHEL KREVANS
SCOTT C. MOORE
ESTHER KIM
MORRISON & FOERSTER LLP
425 Market Street
San Francisco, CA 94105-2482

Counsel for Defendant-Appellee Sling Media, Inc.

June 14, 2013

CERTIFICATE OF INTEREST

Counsel for Defendant-Appellee Sling Media, Inc. certifies the following:

1. The full name of every party represented by me is as follows:

Sling Media, Inc.

2. The name of the real party in interest (if the party named in the caption is not the real party in interest) represented by me is as follows:

N/A.

3. All parent corporations and any publicly held companies that own 10 percent or more of the stock of the party represented by me are as follows:

Sling Media, Inc. is a wholly owned subsidiary of EchoStar Technology L.L.C., which is a wholly owned subsidiary of EchoStar Corporation, a publicly traded company.

Putnam Investments, LLC owns 18.4% of EchoStar Corporation's Class A Stock. Putnam Investments, LLC is a wholly owned subsidiary of Great-West LifeCo Inc., which is a public company traded on the Toronto Stock Exchange.

4. The names of all law firms and partners or associates that appeared for the party now represented by me in the trial court or expected to appear in this Court are as follows:

Morrison & Foerster LLP:
Charles S. Barquist
Esther Kim
Rachel Krevans
Deanne E. Maynard
Scott C. Moore
Brian R. Matsui
Martin Noonen

/s/ Deanne E. Maynard
Deanne E. Maynard

Dated: June 14, 2013

CERTIFICATE OF INTEREST

Counsel for Defendants-Appellees Western Digital Corporation and Western Digital Technologies, Inc. certifies the following:

1. The full name of every party represented by me is as follows:

Western Digital Corporation and Western Digital Technologies, Inc.

2. The name of the real party in interest (if the party named in the caption is not the real party in interest) represented by me is as follows:

N/A.

3. All parent corporations and any publicly held companies that own 10 percent or more of the stock of the party represented by me are as follows:

Western Digital Technologies, Inc. is a wholly owned subsidiary of Western Digital Corporation, which is a publicly traded company.

Hitachi, Ltd owns 10.56% of Western Digital Corporation's shares of common stock. Hitachi, Ltd is a public company traded on the Tokyo Stock Exchange.

4. The names of all law firms and partners or associates that appeared for the party now represented by me in the trial court or expected to appear in this Court are as follows:

King & Kelleher LLP: Dirk van Ausdall, Edward Vincent King, Jr.

Brown White & Newhouse LLP: George Braxton Newhouse, Jr., Thomas Michael Brown

Wiley Rein LLP: Kevin Paul Anderson, Karin Hessler, Robert J. Scheffel

/s/ Kevin P. Anderson
Kevin P. Anderson

Dated: June 14, 2013

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page(s)

CASES

STATEMENT OF RELATED CASES

No other appeal in or from this same civil action was previously before this Court or any other court of appeals.

Because the district court entered judgment under Rule 54(b) of the Federal Rules of Civil Procedure, a case remains pending in the United States District Court of the Northern District of California. *See Nazomi Commc'ns, Inc. v. Nokia Corp.*, No. 10-CV-04686. That case involves plaintiff-appellant Nazomi Communications, Inc. ("Nazomi") and defendants Nokia Corporation and Nokia Inc., defendant Amazon.com, Inc., and defendant Vizio, Inc. As the caption in this appeal reflects, those defendants are not parties to this appeal.

An additional related case is pending in the United States District Court for the Northern District of California: *Nazomi Commc'ns, Inc. v. Samsung Telecomms. Inc.*, No. 10-CV-05545.

Defendant-appellees Sling Media, Inc. ("Sling Media") and Western Digital Corporation and Western Digital Technologies, Inc. (collectively "Western Digital") believe those cases could be affected by or affect this appeal.

STATEMENT OF THE ISSUE

Whether the district court correctly held that Sling Media's Slingbox PRO-HD ("Slingbox") and Western Digital's My Book World Edition ("My Book") do not infringe the asserted claims of Nazomi's U.S. Patent No. 7,080,362 ("the '362 patent") and U.S. Patent No. 7,225,436 ("the '436 patent") because the accused devices are not capable of practicing all the limitations of the asserted claims.

INTRODUCTION

The district court correctly held that the Slingbox and the My Book are not capable of practicing Nazomi's apparatus claims. The asserted claims are directed to a computer processor that uses a hardware accelerator to interpret and convert the popular Java programming language into the native language of the particular processor. A key limitation of every asserted claim is a dedicated hardware accelerator capable of processing Java. Indeed, that was the basis Nazomi used to distinguish its patents from the prior art.

The accused devices are incapable of infringing because neither device has a hardware accelerator capable of processing Java. In fact, Sling Media's Slingbox is not capable of processing any Java programming at all. And it is not even possible to modify the Slingbox to make it capable of infringing. To use Java on the Slingbox, a consumer would need to "hack" the device to manually add Java programming. Even then, the hardware component that Nazomi accuses could not function. Thus, the Slingbox likely would be left useless and unable to function at all. Likewise, undisputed evidence demonstrates Western Digital's My Book is not capable of practicing the functionality that Nazomi claims. Rather than use a hardware accelerator, Western Digital processes Java by using what Nazomi's own patents recognize as prior art. Indeed, the My Book does not use the hardware component Nazomi claims infringes because, among other reasons, Western

Digital could not get it to operate in the My Book. These facts, which Nazomi does not and cannot contest, are fatal to its appeal.

Nazomi nevertheless challenges the ruling below, framing its argument in two ways. Nazomi first claims that the district court imposed functional limitations not present in the asserted claims, and then Nazomi claims that the district court misapplied its construction to the accused devices. But those arguments really boil down to a single contention: because Nazomi purportedly claimed only an apparatus, the accused products can infringe even though they are not "capable of" practicing the claimed functionality.

That argument is contrary to both the asserted claims and this Court's well-settled precedent. When an apparatus claim recites functional language, this Court's rule is clear: "the apparatus as provided must be 'capable' of performing the recited function." *Typhoon Touch Techs., Inc. v. Dell, Inc.*, 659 F.3d 1376, 1380 (Fed. Cir. 2011). It is not enough "that it might later be modified to perform that function." *Id.* For a device to be infringing based on the possibility of subsequent user modification, the device must have been "designed to be altered or assembled before operation." *High Tech Med. Instrumentation, Inc. v. New Image Indus., Inc.*, 49 F.3d 1551, 1556 (Fed. Cir. 1995). In short, the mere presence of hardware elements, without more, is insufficient.

The district court correctly held that this well-established precedent precludes liability on the undisputed facts here. The judgment of non-infringement should be affirmed.

STATEMENT OF FACTS

A. Technology At Issue

Nazomi asserts four apparatus claims of two related patents, claims 48 and 74 of the '362 patent and claims 1 and 5 of the '436 patent.[1] The asserted patents are directed to a central processing unit ("CPU" or processor) that can process two types of computer instruction sets directly in hardware: a register-based instruction set (*e.g.*, instructions in the processor's native language) and a stack-based instruction set (*e.g.*, Java bytecodes).[2]

[1] The patents here descend from a patent this Court twice addressed in rejecting infringement accusations made by Nazomi against the Jazelle circuitry at issue in the present case. *See Nazomi Commc'ns, Inc. v. Arm Holdings, PLC*, 403 F.3d 1364 (Fed. Cir. 2005); *Nazomi Commc'ns, Inc. v. Arm Holdings, PLC*, 266 F. App'x 935 (Fed. Cir. 2008). In the prior litigation, the district court provided two opinions describing the technology at issue. *See Nazomi Commc'ns, Inc. v. Arm Holdings, PLC*, 2003 U.S. Dist. LEXIS 26380 (N.D. Cal. 2003); *Nazomi Commc'ns, Inc. v. Arm Holdings, PLC*, 2006 U.S. Dist. LEXIS 66354 (N.D. Cal. 2006).

[2] Computers use two types of memory organization to store information. Stack-based memories store information on a last-in, first-out basis. Stack-based systems are analogous to a stack of papers in an inbox – to access a paper at the bottom of the stack, a reader must first remove all of the papers above it. In contrast, register-based memories store and retrieve data according to the exact location of each data item. Register-based systems are analogous to an

(Footnote continues on next page.)

1. Background of the technology

Software controls today's computers. A2705. Programmers write software as a series of individual discrete steps that the computer must perform. Software often is written in high-level programming languages, such as C, C++, or Visual Basic. A2699-A2719 at A2705. These languages "use a subset of human vocabulary, such that a person knowledgeable in the basics of computer programming and in the conventions of a particular high-level language can 'read' a program." A2705. The high level programming statements that make up a program are often referred to as the source code for that program. A2705.

CPUs, however, cannot recognize or process high-level programming languages. Before a program can be sent to a processor, the high-level source code must be compiled, or translated, so that the processor can understand what it is being told to do. A2699; A2705-A2706. Most modern CPUs are designed to execute only low-level register-based instruction sets. A2699-A2700; A2709-A2710. To translate high-level source code into a low-level register-based instruction set the CPU can understand, stack-based instructions are read by the decoder within the processor. A2705; A2709-A2710. The decoder then generates

---

(Footnote continued from previous page.)

arrangement of post office boxes, in which the reader identifies and finds the "box" that contains the desired data and is able instantly to retrieve it. A2709.

the correct sets of control signals to carry out the various steps specified by the programmer. A2705-A2706; A2709-A2710.

Typically, a given processor recognizes only a single set of low-level instructions that has been compiled specifically for that type of processor. A2699; A2705-A2706; A2709-A2710. These are commonly called the "native instructions" for the particular processor. A2700. Each CPU uses its own register-based native instructions. A2700. For example, Intel-based processors use one set of native instructions, Macintosh PowerPC-based processors use a different set of native instructions, and ARM processors use another still. *See* A2706. Thus, high-level source code compiled into register-based instructions for one type of CPU does not work for other types of CPUs. A2706.

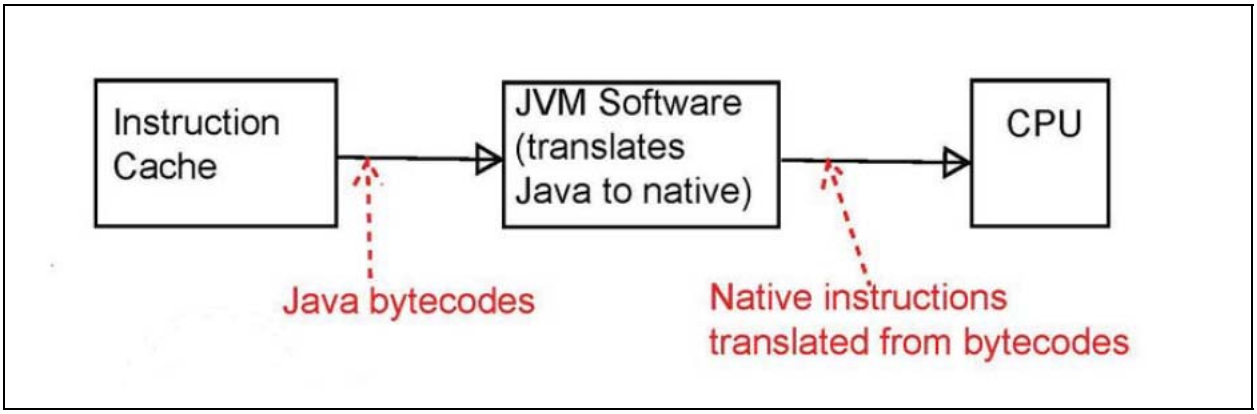
2. Java programming

Java is a high-level programming language that allows programmers to write a single program that can run on a variety of different processors, without recompiling the software for each processor. A2699-A2700; A2706. Like other high-level programming, Java must be compiled into low-level instructions called Java bytecodes before it can be used. A96; A125; A2699-A2700; A2706; A2709-A2710.

Java bytecodes are not "register-based" instructions; Java bytecodes are a different type of low-level instruction set called "stack-based" instructions. Java

bytecodes are device-independent – they are not compiled to run on particular processors. However, these low-level Java bytecodes cannot be executed directly on most processors. A96; A125; A2699-A2700; A2706; A2709-A2710.

To execute Java bytecodes, a user typically must have an intermediate software program called a Java Virtual Machine ("JVM") running on the computer. A2700; A2706; A2710. The JVM software takes the stack-based Java bytecodes and translates them into the particular register-based instructions that are native to the processor inside the user's computer. A2700; A2706; A2710. This allows Java bytecodes to be platform independent – the same Java bytecodes can be run on any computer, regardless of the type of processor, so long as the computer includes JVM software capable of translating the stack-based Java bytecodes into register-based instructions native to that computer's processor. A2699-A2700; A2705-A2706; A2709-A2710. The process of executing stack-based Java bytecodes on a user's computer using JVM software conceptually works as follows:



Using JVM software, however, is not without disadvantage: JVM software adds processing steps, reduces the overall execution speed of a program, and can require additional memory. A96; A125; A2700; A2706; A2710.

3. The asserted patents claim a hardware accelerator that can process Java bytecodes

Nazomi's two asserted patents seek to avoid the disadvantage of prior art JVM software. A84-A102 ('362 patent); A103-A132 ('436 patent). These related patents claim a CPU that can process both stack-based and register-based instructions directly in the hardware of the CPU. A84-A102 ('362 patent) at A97 (col.3:31-37), A100-A101 (col.10:57-col.11:6), A101 (col.12:29-47); A103-A132 ('436 patent) at A126 (col.4:23-29), A131 (col.14:30-46), A132 (col.15:27-50). Both patents disclose a "Java hardware accelerator" that translates stack-based Java bytecode instructions into register-based "native" instructions that are usable by a register-based CPU. A84-A102 ('362 patent) at A96 (col.2:10-17, col.2:23-25), A97 (col.4:6-27), A98 (col.5:28-29, col.5:64-66); A103-A132 ('436 patent) at A125 (col.2:6-8, col.2:44-48), A126 (col.4:16-19). This is shown in Figure 1 and the associated descriptions in Nazomi's patents:

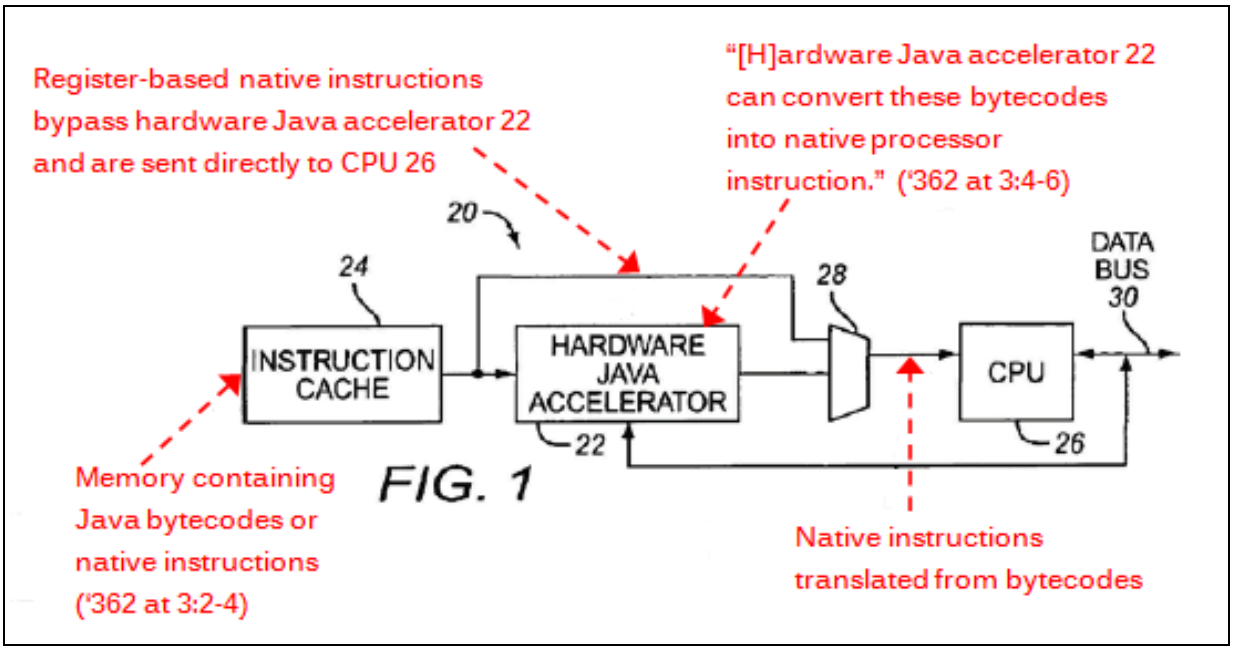


FIG. 1

A84-A102 ('362 patent) at A87 (fig.1), A96-A97 (col.2:59-col.3:45) (text associated with the dashed lines added); A103-A132 ('436 patent) at A106 (fig.1), A126 (col.3:49-col.4:36) (same).

In Nazomi's claimed apparatus, the conversion of the Java bytecodes into native CPU instructions has been moved (at least in part) from the JVM software to the "hardware Java accelerator" (22 in Figure 1 above). Register-based native instructions bypass the hardware Java accelerator and are sent directly to the CPU along path 20 in Figure 1 above. Java bytecodes are routed to the hardware Java accelerator (22 in Figure 1 above), which translates these stack-based instructions to native register-based instructions. All instructions – whether register-based or originally native stack-based – may now be executed by the CPU (26 in Figure 1 above).

All four asserted claims contain functional language directed at the processing of both stack-based and register-based instructions in hardware. Claim 1 of the '436 patent recites:

> 1. A CPU for executing stack and register-based instructions, comprising:
>
> execute logic for executing the register-based instructions;
>
> a register file associated with the executed logic; and
>
> a hardware accelerator to process stack-based instructions in cooperation with the execute logic, wherein the hardware accelerator generates a new virtual machine program counter (PC) due to a "jump subroutine JSR" or "jump subroutine wide JSR_W" bytecode by sign extending the immediate branch offset following the "jump subroutine JSR" or "jump subrountine [sic] wide JSR_W" byte code and adding it to the virtual machine (PC) of the current byte code instruction, computes the return virtual machine program counter and pushes the return virtual machine counter[.]

A131 (col.14:30-46). Claim 5 recites:

> 5. A central processing unit (CPU) comprising:
>
> execute logic to receive and process input corresponding to register-based instruction;
>
> a hardware accelerator to process stack-based instructions to produce an output that can be processed by the execute logic;

> an operand stack for the stack-based instructions, the operand stack being maintained in a register file as a ring buffer;
>
> an overflow/underflow mechanism for moving operands in the operand stack between a register file and a memory, said register file also storing data associated with the register-based instructions;
>
> a bytecode buffer that receives stack-based instructions from the memory; and
>
> an instruction decode unit coupled to the bytecode buffer to decode instructions received from the bytecode buffer and to provide an indication of how many bytes have been processed; and
>
> a common program counter for the stack-based instructions and the register-based instructions, wherein the common program counter is incremented by the indication of the number of bytes processed.

A132 (col.15:26-50).

Similarly, claim 48 of the ’362 patent recites:

> 48. A central processing unit (CPU), capable of executing a plurality of instruction sets comprising:
>
> an execution unit and associated register file, the execution unit to execute instructions of a plurality of instruction sets, including a stack-based and a register-based instruction set;
>
> a mechanism to maintain at least some data for the plurality of instruction sets in the register file including maintaining an operand stack for the stack-based instructions in the register file and an indication of a depth of the operand stack;

> a stack control mechanism that includes at least one of an overflow and underflow mechanism, wherein at least some of the operands are moved between the register file and memory; and
>
> a mechanism to generate an exception in respect of selected stack-based instructions.

A100-A101 (col.10:57-11:6). And claim 74 of the '362 patent recites:

> 74. A central processing unit (CPU) comprising:
>
> a decoding mechanism to decode instructions of a plurality of instruction sets including a -stack-based [sic] instruction set and a register-based instruction set;
>
> a register file, wherein an operand stack to store operands associated with instructions of the stack-based instruction set is maintained; and wherein data associated with instructions of the register-based instruction set is maintained;
>
> at least one of an overflow and underflow mechanism to cause the operand to—be moved between the register file and memory; and
>
> an execution unit that processes the output of the decoding of the instructions of the stack-based instruction set, and the decoding of the instructions of the register-based instruction set, including processing exceptions in respect of selected instructions of the stack-based instruction set within a virtual machine.

A101 (col.12:29-47). In short, the four claims that Nazomi asserts all require a hardware accelerator that can function to process both Java bytecodes and native instructions.

The specifications in both patents further highlight that a hardware accelerator to process stack-based instructions is critical to the claims. Indeed, that was the basis on which the specification distinguished Nazomi's invention from the prior art. The specification states that a disadvantage of using the JVM – which used software to translate Java bytecodes into a CPU's native register-based instructions – was slow execution speeds. A96 (col.1:42-46); A125 (col.1:42-48). In the prior art, the software-based translation of Java bytecodes could be sped up with faster (and more expensive and energy-consuming) processors or by using certain techniques that "result[ed] in additional memory overhead." A96 (col.1:54, 61-64); *see* A125 (col.1:53-54). But to do so would "go[] against the consumer application requirements of low cost and low power." A96 (col.1:66-67); A125 (col.1:66-67).

The specifications for both asserted patents teach Nazomi's solution to this problem found in the prior art, namely, Nazomi's use of a hardware accelerator to translate Java bytecodes into the CPU's native instructions. The specifications state that, unlike the prior art, "at least part of the Java™ Virtual Machine is implemented in hardware as the Java™ hardware accelerator." A96 (col.2:15-17); *see* A96 (col.2:65-67) ("the translation of the Java™ bytecodes into native processor instructions is at least partially done in the hardware Java™ accelerator"). And "[t]he hardware Java™ accelerator can convert the stack-based

Java™ bytecodes into a [sic] register-based native instructions on a CPU." A96 (col.2:23-25). A CPU that is not capable of decoding Java bytecodes through a hardware accelerator is indistinguishable from the prior art. A96 (col.2:10-14).

B. Architecture And Operation Of The ARM Processor

ARM develops and licenses processor core designs. Other companies, such as Qualcomm and Texas Instruments, use ARM's core designs to build complete CPUs. *See* A3611-A3622 at A3612-A3613 ¶5. These Qualcomm and Texas Instruments CPUs then are incorporated into devices, such as the Slingbox and the My Book. One of the many components in these CPUs is the ARM-designed 926EJ-S processer core. A2076-A2083 at A2079-A2080 ¶24; A2293-A2310 at A2296 ¶24.

ARM has developed an improvement to its processor core designs that increases the speed at which Java programs are processed. ARM improved the processing speed through a combination of hardware and software. ARM marketed this improvement under the name "Jazelle." The Jazelle hardware component is a portion of the ARM 926EJ-S processor core. A2722 ¶5; A3613 ¶8, A3614 ¶16.

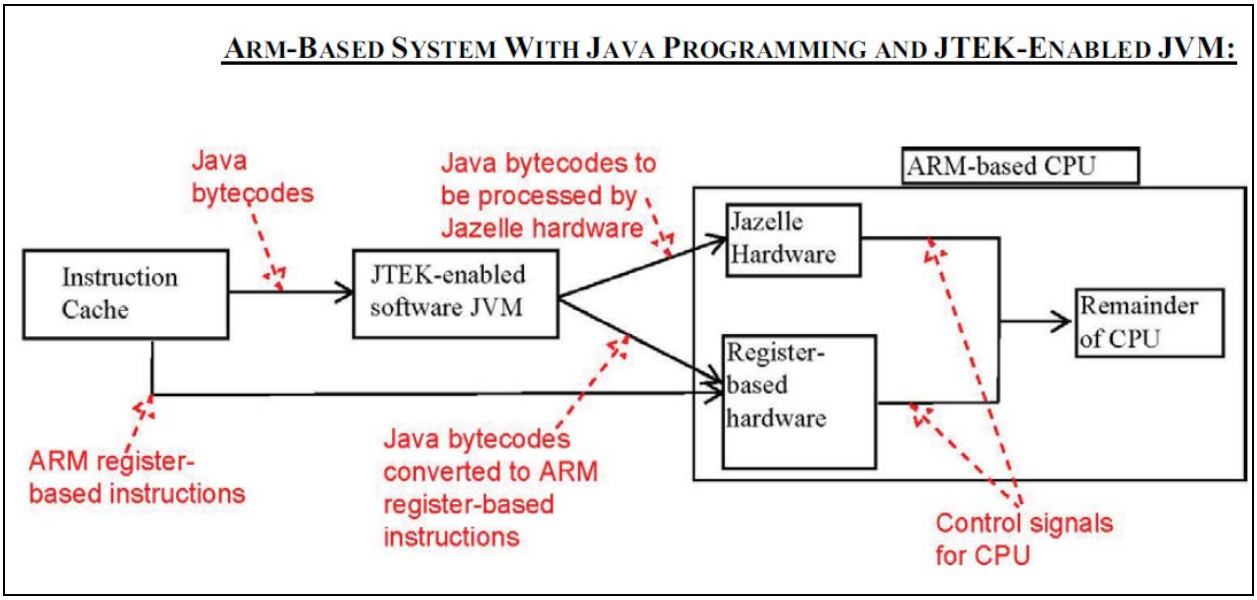
The Jazelle hardware is not enabled in ARM 926EJ-S processor cores. If a manufacturer wishes to make and sell products that use Jazelle, the manufacturer must include additional technology that modifies hardware registers in the Jazelle

hardware component of the ARM 926EJ-S core. This requires installing the Java Technology Enabling Kit ("JTEK"), a technology package licensed by ARM. Manufacturers that do not purchase and install JTEK cannot use the Jazelle portion of ARM processors. A3614-A3615 ¶¶16-17; A3890 ¶¶4-5.

Enabling the Jazelle portion of the ARM 926EJ-S processor core is not simple; it requires not just the addition of JTEK software but also several complicated, proprietary steps. A3614 ¶16. ARM discloses those steps only to licensees who separately have licensed JTEK. A3614 ¶16. The JTEK technology is incorporated only into certain JVMs, which are provided only to companies that have licensed JTEK. A2722 ¶6. These JVMs are referred to as "JTEK-enabled JVM[s]." A2722 ¶6. If an ARM 926EJ-S processor core is used in a product that does not include a JTEK-enabled JVM, the Jazelle hardware within the ARM processor does not function, and is not capable of being used at all – much less to process Java bytecodes. A3614 ¶16; A2722-A2724 ¶¶5-8.

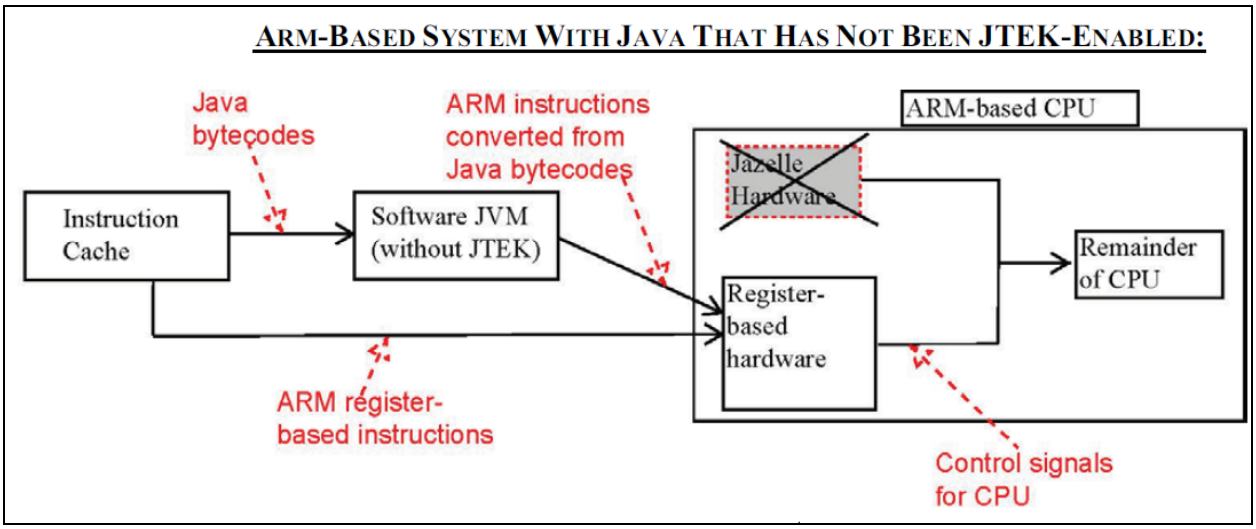
If a manufacturer produces an ARM-based system that includes a JTEK-enabled ARM processor, Jazelle potentially may be available for use. A2723-A2724 ¶6. In such a processor, some Java bytecodes are processed by software in the JTEK-enabled software JVM and other Java bytecodes are processed by the Jazelle hardware if Jazelle is selected for use. As shown below, some of the stack-based Java bytecodes are routed through the Jazelle hardware where they are

processed directly into control signals, A3620-A3622 ¶¶39-41, and other Java bytecodes are translated by software in the JVM (as in the prior art) into register-based native ARM instructions, A3620-A3622 ¶¶39-41:



A2723 ¶6.

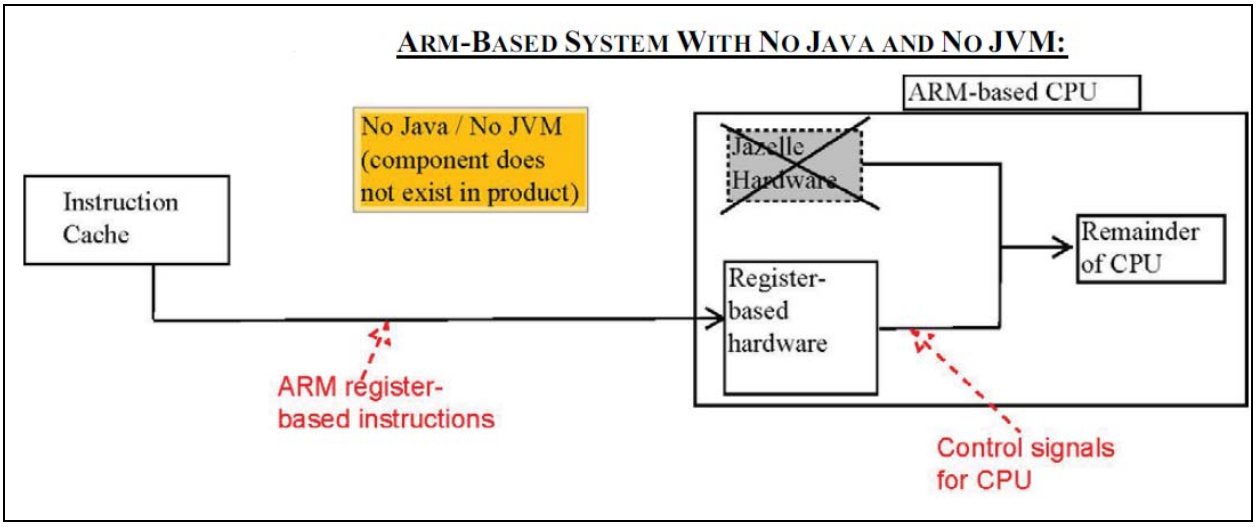
However, when a manufacturer such as Western Digital uses an ARM-based system that uses Java programming and includes an ARM 926EJ-S processor core, but does *not* license JTEK software from ARM (so Jazelle does not function), *all* stack-based Java bytecodes are translated by a software JVM (just as in the prior art). As shown below, the software JVM translates the Java bytecodes into ARM register-based instructions, which then are processed just as any other ARM register-based instructions are processed. A2723 ¶7.



A2723 ¶7.

When JTEK is not installed (so the Jazelle hardware is disabled and dormant in a device), the Jazelle hardware is not capable of processing any instructions. A2723 ¶7. Instead, the system behaves just as software JVMs have behaved since long before Nazomi's claimed invention. A3618-A3619 ¶¶29-32. All stack-based Java bytecodes are converted to register-based native instructions by the software JVM, and only native register-based ARM instructions are sent to the processor. A3618 ¶¶30-31. When Jazelle was initially developed, ARM processors using a standard JVM (without JTEK) executed Java programs more slowly than those using the Jazelle hardware because the JVM had to interpret the Java bytecodes in software before the processor could execute the interpreted ARM instructions. A3619 ¶32. By the time Western Digital developed the My Book, however, the software JVM technology had improved sufficiently that it met, or exceeded, the performance available from Jazelle.

Finally, no stack-based Java bytecodes are processed at all when a manufacturer such as Sling Media uses an ARM-based system that (i) includes an ARM 926EJ-S processor core, (ii) does not install JTEK so Jazelle does not function, (iii) and does not use Java programming and therefore only processes register-based instructions. A2724; A4976-A4996 at A4981-A4982. As shown below, only register-based instructions are processed, no JVM is incorporated, and the Jazelle hardware does not process anything.



A2724.

C. The Accused Devices Are Not Capable Of Processing Java Bytecodes In Hardware

1. Sling Media's Slingbox

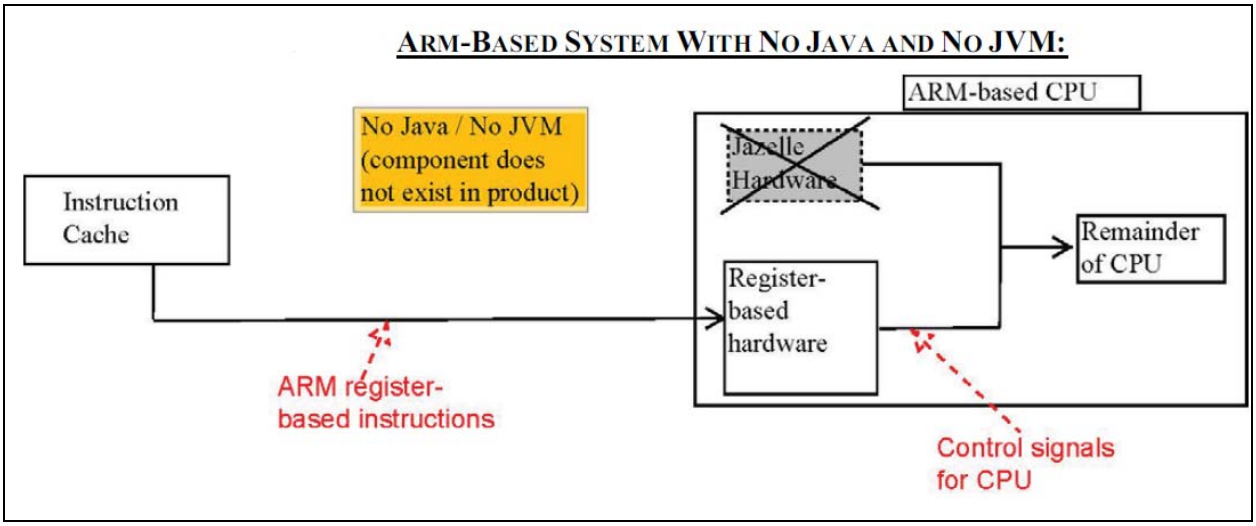
Nazomi alleged infringement by one Sling Media product: the Slingbox. A2572-A2662. The Slingbox uses a Texas Instruments Da Vinci processor, which includes among its components an ARM 926EJ-S processor core. A3886-A3887 ¶6.

Sling Media has not licensed or installed ARM's JTEK, and Sling Media is not capable of enabling the Jazelle circuitry in manufacturing. A3889-A3890 ¶¶4-5. Thus, the Jazelle hardware within the ARM 926EJ-S processor is not capable of doing anything. A3886 ¶6. It is not capable of processing any instructions, including the stack-based instructions recited in the asserted claims. A2723-24 ¶8. In fact, the Slingbox does not even contain any Java bytecodes – the only type of stack-based instructions that Jazelle can process. A3890 ¶¶4-5; A3886-A3887 ¶¶2-6; A4981-A4982; *see* A2723-A2724 ¶8. Because the Slingbox processes no Java, it has no JVM at all.

Finally, the Slingbox is not susceptible of modification to enable the processing of Java or the Jazelle hardware component of the ARM processor. Nothing on the Slingbox permits a user to download or store stack-based instructions on the device. A3886 ¶¶2-4. The only conceivable way a user could install Java programing on a Slingbox would be to "hack" the device and manually add Java programming to the software stored on the device. A3886 ¶4. That is not authorized by Sling Media. A3886-A3887 ¶6. Indeed, it likely would render the device useless. A3886 ¶4. And Sling Media is aware of no instance in which someone has tried to "hack" a Slingbox in this manner. A3887 ¶6. Even then, the Jazelle hardware would remain inoperable because Sling Media did not license or

install ARM's JTEK software, and thus did not activate Jazelle during manufacturing. A3890 ¶¶2-5.

The type of ARM-based system the Slingbox uses thus is one where *no* Java bytecodes are processed. There is no JVM in the device. A4981-A4982. The Jazelle hardware accelerator is not capable of processing any stack-based Java bytecodes because only register-based instructions are processed by the CPU:



A2724; A3886-A3887 ¶6.

2. Western Digital's My Book

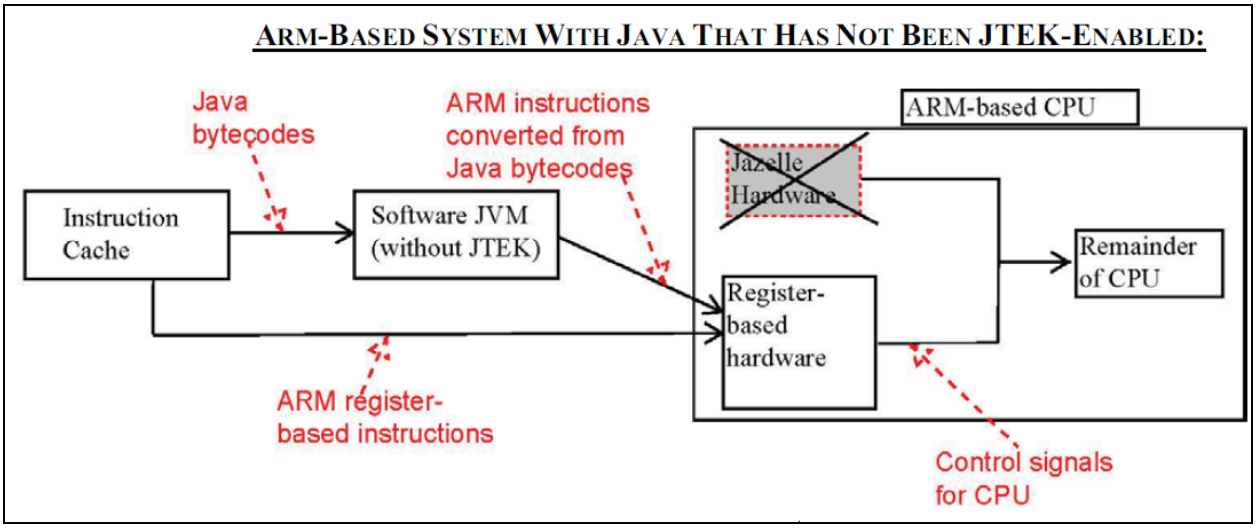
Nazomi also accused a single Western Digital product: the My Book, which is a network-attached disk drive that can be accessed through a computer network to store information. A2484-A2570. The My Book incorporates Java Virtual Machine from Sun Microsystems, which includes an ARM 926EJ-S processor core, but is not JTEK-enabled. A2670-A2693 at A2675-A2679, A2687; A2744-A2746.

The My Book processes no stack-based instructions in hardware. It only uses software (specifically, a JVM using Just-in-Time ("JIT") conversion) to convert Java bytecodes. A2745; A3717-3756 at A3723, A3724 & A3730. As sold to customers, the My Book does not enable or use Jazelle in any fashion, and cannot reasonably be modified to use Jazelle. A2724; A2745-A2746 ¶¶3-11; A3735-A3756. Just like Sling Media's product, Western Digital has never licensed ARM's JTEK software. A3724; A3735-A3736; A3743.

Moreover, the My Book cannot reasonably be modified to use Jazelle. To modify the product, the customer would have to "hack" the Western Digital product to incorporate all new software. A3745. The customer first would have to modify JVM software that is hidden from the user, as the "Java Virtual Machine is located on the hidden portions of the disk drive." A3750. Western Digital does not authorize such a modification. A3750. Even if the customer could locate the "hidden" software, the user either would have to replace all the software that processes Java or would have to know exactly which specific files to replace. A3755. To Western Digital's knowledge, no one has ever modified the software so as to be able to use Jazelle. A3741; A3745-A3756. Moreover, even if someone were able to modify the software to try to use Jazelle, the Jazelle circuitry within the Western Digital product still might not function. In fact, Western Digital could

never get the Jazelle hardware to function successfully, even when Western Digital attempted to use Jazelle in some of its prototypes. A3729-A3730.

The type of ARM-based system Western Digital's My Book uses is indistinguishable from the prior art – the software JVM translates the Java bytecodes into ARM register-based instructions, which then are processed just as any other ARM register-based instructions are processed:



A2723 ¶7, A2745 ¶3.

D. District Court Proceedings

Nazomi filed this suit against a number of defendants. As to Sling Media and Western Digital, Nazomi alleged that the Jazelle component of the ARM 926EJ-S processor core in the Slingbox and the My Book infringes the four asserted claims. A137 ¶¶26, 28.

1. Sling Media and Western Digital's first summary judgment motion

Sling Media and Western Digital immediately moved for summary judgment of non-infringement because the Slingbox and the My Book do not use, and are not capable of using, the hardware Java accelerator that Nazomi alleges infringes. A2429-A2757, A2777-A2792, A2805-A2829, A2855-A2860. Nazomi opposed by seeking discovery pursuant to Rule 56(d) of the Federal Rules of Civil Procedure. A2855.

The district court denied the summary judgment motion without prejudice. A2860. The court concluded that resolution of the motion would require claim construction, which the court had not yet resolved. A2857-A2858. The district court further held that, "[e]ven under a settled claim construction, determining whether an accused infringing device is 'capable of' performing the recited function is a fact-intensive inquiry." A2858. The court noted Nazomi's assertion that it needed "the source code for defendants' products because the 'operation of this source code is foundational to' defendants' argument that the Jazelle hardware is disabled or dormant in the products." A2859 (quoting Nazomi's opposition to summary judgment motion). The court also noted Nazomi's claimed need of depositions of the parties' witnesses, as well as depositions and discovery from third-party chipset manufacturers. A2859.

2. Second summary judgment motion

After the district court granted Nazomi discovery, Sling Media immediately made its witnesses available for deposition. A3881-A3883 at A3882 ¶3. Sling Media also made its source code available for inspection and obtained permission from its CPU manufacturer, Texas Instruments, for Nazomi to inspect the Texas Instruments-supplied source code used by the Slingbox. A3882-A3883 ¶¶4-7. While Nazomi did inspect the source code on two occasions, it never responded to Sling Media's offer of depositions. A3882-A3883 ¶¶5, 7-8.

Western Digital provided extensive discovery to Nazomi with respect to the My Book accused product. Western Digital also made its source code available for inspection by Nazomi. Nazomi then took Rule 30(b)(6) depositions of Western Digital, Western Digital's chip supplier (PLX Technology (formerly known as Oxford)), and ARM. These depositions related to the specific technology at issue in Western Digital's product. A3557-A3558 ¶¶7-9.

Sling Media and Western Digital renewed their summary judgment motion. A3517-A3890; A3897-A4507; A4916-A4997. Following a technology tutorial at which both sides provided extensive presentations regarding the technology in the patents and the accused products (A5015-A5096) and oral arguments on the pending summary judgment motion (A5097), the district court granted the motion. A1-A10.

3. District court's decision

Claim construction. In granting the second summary judgment motion, the district court construed the claims. Nazomi argued that its "apparatus claims are directed to physical circuitry in CPU hardware that is configured to perform the recited functional language." A4728. The district court construed the "asserted claims to cover a hardware apparatus configured to process Java bytecodes without modification by an end-user." A8.

The court reasoned that the structural elements of all the asserted claims are defined by their functionality. A5. The court noted that the "specifications also consistently describe the invention as a hardware apparatus used to process Java bytecodes." A6. "The specifications further emphasize that the primary advantage of the patented invention over the prior art is that the processing of stack-based instructions in hardware, rather than software, results in 'an improved system for implementing *Java™ programs*.'" A6 (quoting '362 patent, A96 (col.2:1-2); '436 patent, A125 (col.2:1-2)) (emphasis added by district court). The court concluded that "[b]y defining structural elements in functional terms and highlighting the implementation of Java-based programming, Nazomi plainly added limitations that would not be present had it used generic, structural language." A6.

The court nevertheless declined to adopt defendants' construction, which would have limited the claims to "processors that actually perform the recited

functions when placed in operation." A6-A7 (internal quotation omitted). Instead, the court "agree[d] with Nazomi that the functional language of the asserted claims recites capabilities of the claimed structures." A7.

As the district court explained, "'[c]apability,' however, is not a blank check." A7. Citing decisions of this Court, the district court noted: "[a]s the Federal Circuit has made clear, the scope of an apparatus claim phrased in functional terms is generally limited to products 'configured' to perform the recited functions without 'modification' by the end-user." A7. The district court reasoned that "[t]he emphasis on an accused product's default configuration is consistent with the language of the patents-in-suit, which describes an invention designed primarily for a single purpose – processing Java bytecodes – rather than operation in multiple modes or customization by end-users." A7-A8. "Accordingly, the court construe[d] the asserted claims to cover a hardware apparatus configured to process Java bytecodes without modification by an end-user." A8.

Non-infringement. As to infringement, the district court noted that it "is undisputed that Jazelle cannot process Java bytecodes in the absence of JTEK software." A8. "It also is undisputed that the accused products do not use a JTEK-enabled JVM." A8. "The accused products are thus not configured to process Java bytecodes in hardware because their Jazelle circuitry is 'dormant' by default." A8.

"Further, it is clear that the moving defendants did not 'intend[] or anticipate[]'" their products to be configured in an infringing manner." A8-A9 (quoting *High Tech Med. Instrumentation*, 49 F.3d at 1555) (alteration by district court). "Neither product is designed to allow a user to download software that would 'activate' Jazelle." A9.

The court noted that Western Digital "produced uncontroverted testimony that a hardware-based JVM 'was not stable enough' for use on the MyBook, and that it elected to employ a software-based solution instead." A9. The court also explained that the "Slingbox does not use Java programs at all, and thus has no need for *any* JVM, let alone one that utilizes Jazelle." A9 (emphasis in original).

The district court rejected Nazomi's argument "that the mere presence of Jazelle circuitry is sufficient to find infringement." A9. In so doing, the court distinguished the undisputed facts here from situations where an accused product is designed so that a user can enable the claimed function without having to modify the product. A10 (discussing *Silicon Graphics, Inc. v. ATI Techs., Inc.*, 607 F.3d 784, 794 (Fed. Cir. 2010)). Here, as the district court held, "the MyBook and Slingbox cannot process Java bytecodes in hardware when sold to consumers, nor can consumers enable Jazelle without first modifying the accused devices in a manner not contemplated by the moving defendants." A10.

4. Rule 54(b) judgment

The district court issued final judgment in favor of defendants Sling Media and Western Digital under Rule 54(b). A11-A16.

SUMMARY OF ARGUMENT

I. This Court's rule is well-settled. When an apparatus claim recites functional language, an accused device as provided must be capable of performing that claimed functionality to infringe. It is not enough that a user could make an unauthorized modification to the device that would be infringing. Although Nazomi highlights decisions where the accused device ultimately was held to infringe, those decisions did not espouse a different rule. Rather, those decisions involved facts showing that the accused devices had the required capabilities – facts that are not present here.

II. Applying this Court's well-settled rule to the undisputed facts of this case, the district court correctly held that the Slingbox and the My Book do not infringe the asserted claims.

A. Nazomi's apparatus claims are defined by their functional language, which Nazomi cannot now avoid when asserting its patents against others. Nazomi did not claim a generic CPU. Instead, each asserted claim recites a hardware accelerator that can process stack-based instructions. The claims require that functionality to be available to the end user in the device as provided. The claims

do not encompass the possibility of subsequent user modification. Indeed, the specification demonstrates that a hardware accelerator processing stack-based instructions was a key distinguishing feature from the prior art.

B. The Slingbox and the My Book are incapable of infringing the asserted claims.

1. The Slingbox is not capable of processing stack-based instructions in a hardware accelerator. The CPU incorporated in the Slingbox is not capable of processing stack-based instructions in hardware. Indeed, Sling Media's device does not process stack-based instructions at all. On that basis alone, the judgment in favor of Sling Media should be affirmed.

Moreover, there is no record evidence that the Slingbox is intended to be – or ever could be – modified in a manner to infringe. Any attempted modification would require an end user to "hack" the device, which likely would render it inoperable and incapable of functioning as intended.

2. Similarly, the My Book does not infringe Nazomi's asserted claims. No stack-based instructions are processed in a hardware accelerator. Indeed, Western Digital concluded that such hardware lacked the stability for the device to function properly. Thus, like the Slingbox, the My Book's CPU is not capable of processing stack-based instructions in hardware. Nor can the device be reasonably modified to use a hardware accelerator. An end user would have to "hack"

Western Digital's My Book, make unauthorized modifications to the device, and even then, there is no basis to believe such an unauthorized modification would work.

C. Nazomi contends that the district court should have focused on circuitry within the devices in isolation, rather than on the devices as a whole. But this Court consistently examines whether the accused device practices the claim. In any event, neither the Slingbox nor the My Book – whether considered as a whole or focusing solely on the CPUs (or the processor cores inside the CPUs) – meets all the limitations of Nazomi's claims. And Nazomi cites no case where infringement was based on an allegation that some internal part of a device could be modified post-sale in an unauthorized way so as to infringe.

D. Finally, Nazomi asserts that the district court erroneously imported an intent requirement into its infringement analysis. Nazomi is wrong. Under this Court's precedent, when an accused device does not meet the limitations of the asserted claims, the device may nevertheless infringe if the defendant designed the device with an infringing, post-sale modification in mind. Here, because neither the Slingbox nor the My Book infringe as sold to or as usable by end users, the district court did not err by examining whether the devices were intended to be modified post-sale in an infringing manner. It correctly concluded that they were not, a conclusion Nazomi does not challenge.

STANDARD OF REVIEW

The Court reviews the grant of summary judgment de novo. *Nystrom v. Trex Co.*, 424 F.3d 1136, 1141 (Fed. Cir. 2005). Claim construction is a question of law that this Court reviews de novo. *Meyer Intellectual Props. Ltd. v. Bodum, Inc.*, 690 F.3d 1354, 1368 (Fed. Cir. 2012).

ARGUMENT

Nazomi challenges the district court's decision by raising one principal argument framed two different ways, first as a claim construction challenge and then as a challenge to the non-infringement holding. Nazomi's argument fails under this Court's precedent regardless of how it is framed.

I. UNDER THIS COURT'S PRECEDENT, TO INFRINGE AN APPARATUS CLAIM THAT RECITES FUNCTIONAL LANGUAGE, AN ACCUSED PRODUCT MUST BE CAPABLE OF PERFORMING THE CLAIMED FUNCTIONALITY WITHOUT UNAUTHORIZED USER MODIFICATION

The rule in this Court is well settled: to infringe a claim reciting functional language, an accused device must be capable of performing the claimed functionality as provided, without the user having to modify it. *Typhoon Touch*, 659 F.3d at 1380; *Finjan, Inc. v. Secure Computing Corp.*, 626 F.3d 1197, 1205 (Fed. Cir. 2010); *Silicon Graphics*, 607 F.3d at 794. And for a manufacturer or seller of a device to be liable for infringement based on the possibility of subsequent user modification, the accused device must have been "designed to be

altered or assembled before operation." *High Tech Med. Instrumentation*, 49 F.3d at 1556.

And, as shown below, this Court consistently applies its rule. While Nazomi highlights decisions ultimately finding infringement, those outcomes turned not on conflicting rules of law but on the different facts of those cases. They do not help Nazomi on the undisputed facts here. *See* Nazomi Br., 17-19, 21, 26-34, 39-46.

As the district court recognized (A8-A9), this Court applied this legal rule in *High Tech Medical Instrumentation.* In that preliminary injunction appeal, the Court held that an accused device – an endoscope camera used for dental work – did not likely infringe. *High Tech Med. Instrumentation*, 49 F.3d at 1553. The asserted claim recited "a camera disposed in said body member, said camera being rotatably coupled to said body member." *Id.* at 1553. According to the specification, "a camera that is 'rotatably coupled' to its housing permits the operator to rotate the camera within the housing during use while the camera remains at the proper distance" to remain in focus. *Id.*

As sold to end users, the accused device in *High Tech Medical Instrumentation* did not rotate. Specifically, the device included a "camera [that] was not intended to rotate within the housing" and there were "set screws [to] prevent such rotation." *Id.* at 1555. The device could be modified, however, by loosening the set screws, so that the camera could rotate in its housing. *Id.*

Because the device could be modified to rotate, the district court had held that the accused device was "capable of being rotated" in an infringing manner. *Id.* This Court reversed. *Id.* at 1558.

This Court noted that to meet the "rotatably coupled" claim limitation, the "original and intended operating configuration of the device must be altered – by loosening the set screws – in order for the camera to rotate." *Id.* at 1555. The Court held that was insufficient for infringement: "a device does not infringe simply because it is possible to alter it in a way that would satisfy all the limitations of a patent claim." *Id.* (citing *Hap Corp. v. Heyman Mfg. Co.*, 311 F.2d 839, 843 (1st Cir. 1962)). And finding no reason in the record to conclude that the device was "designed to be altered or assembled before operation," the Court reversed the district court. *Id.* at 1555-56, 1558.

This Court again applied these legal principles in *Telemac Cellular Corporation v. Topp Telecom, Inc.*, 247 F.3d 1316 (Fed. Cir. 2001). The asserted patent there claimed a "phone system including a mobile phone having internal accounting capabilities." *Id.* at 1319. In particular, the asserted claim included a "complex billing algorithm" that was construed to require the "storage of phone rates for, at a minimum, local, long distance, international, and roaming calls." *Id.* at 1326. The accused system, however, did not permit users to place international calls. *Id.* at 1330.

The patentee nevertheless asserted that "even though Topp has chosen not to permit direct dialing of international calls, the capability of billing for international rates is nonetheless present in the phone's source code." *Id.* The patentee argued that because the accused "system is capable of being modified to place, and charge for, international calls, Topp's system infringes." *Id.* This Court rejected that argument: "Under the precedent of this circuit, however, that a device is capable of being modified to operate in an infringing manner is not sufficient, by itself, to support a finding of infringement." *Id.* This Court thus found no infringement because, "due to a restriction built into the software program stored in the telephone's memory, a user of [defendant's] system is prevented from directly placing international calls." *Id.*

Nazomi attempts to distinguish *Telemac Cellular*, contending in a footnote that the claims there required "actual[] use" as opposed to capability. Nazomi Br. 41 n.8 (emphasis omitted). That is not so. This Court did not require that the accused device actually be used to make international calls; instead, the device had to be capable of making such calls in the manner recited by the claims.[3]

[3] Moreover, Nazomi points to the decision's doctrine-of-equivalents analysis – an issue not present in this appeal – not its "capable of" infringement analysis. *See Telemac Cellular*, 247 F.3d at 1330-31. That equivalents inquiry examined whether the *unmodified* device "perform[ed] substantially the same function in substantially the same way to achieve substantially the same result." *Id.* at 1331. The Court concluded that the mere ability to place international calls indirectly

(Footnote continues on next page.)

This Court applied the same legal rule yet again in *Typhoon Touch*. Indeed, the Court rejected the very argument Nazomi presses here. Like Nazomi, the patentee in *Typhoon Touch* argued that the district court incorrectly had read a "use" limitation into its apparatus claims. Nazomi Br. 17; *Typhoon Touch* 659 F.3d at 1380-1381.

Typhoon Touch involved patents describing the deficiencies of a portable computer that requires a keyboard for data entry and the advantages of a portable system using a touchscreen. The designated representative claim recited structures with functional language. One limitation required "a memory for storing at least one data collection application configured to determine contents and formats of said inquiries displayed on said screen." *Id.* at 1379 (emphasis omitted). Another limitation required a "run-time utility operating in conjunction with said processor to execute said application and said libraries to facilitate data collection operations." *Id.* at 1379-1380 (emphasis omitted).

The patentee Typhoon contended that it was sufficient for the accused device to have "the capability of being configured or programmed to perform the stated function, although not so structured in the device provided by a defendant."

(Footnote continued from previous page.)

(through an outside international carrier) did not satisfy the equivalents standard because "no international rate is stored in the phone and no charges are calculated using an international rate." *Id.* at 1331.

Id. at 1380. Like Nazomi, Typhoon emphasized that its claims were apparatus claims, arguing "that it is irrelevant if the function is actually performed by the device, if the device can be programmed or configured to perform the function." *Id.*

This Court disagreed. The Court reiterated the legal rule "that the apparatus as provided must be 'capable' of performing the recited functions, not that it might later be modified to perform that function." *Id.* The Court thus held that the functional "memory for storing" language of the claims "requires that the memory is actually programmed or configured to store the data collection application." *Id.* at 1381. Likewise, the Court held that the "operating in conjunction" clause "requires that the device is programmed or configured to perform the stated function." *Id.* at 1382.

Nazomi attempts to limit *Typhoon Touch*, arguing it depended on the particular claim language "configured to." Nazomi Br. 30-31. But this Court placed no significance on those two words in particular. *Typhoon Touch*, 659 F.3d at 1379. Indeed, only the "memory for storing" limitation included "configured to"; the other three limitations did not have that language. Nevertheless, the Court held that, to infringe, the accused device would have to be configured to practice the functionality in each of those limitations. *Id.* at 1381-1382. Thus, with respect to the "operating in conjunction" limitation, this Court rejected Typhoon's

argument that "it suffices if the computer-implemented structures can be configured to operate in conjunction with each other, whether or not they have been so configured in the device." *Id.*

Nazomi acknowledges that only one of the *Typhoon Touch* limitations included "configured to," but argues that the constructions of the other three limitations were supported by intrinsic evidence. Nazomi Br. 31 n.6. As shown above (*see* pp. 8-14 *supra*), the intrinsic record requiring the functional limitations here is just as strong, if not stronger. The district court did not err by requiring Nazomi's claims to "conform[] with the inventors' description of what they invented." *Typhoon Touch*, 659 F.3d at 1382.[4]

Nor did the district court misinterpret *Fantasy Sports*, as Nazomi argues at pages 27 to 28 of its brief. That decision also applies this Court's well-settled legal rule, and the district court correctly concluded that it does not help Nazomi on the

[4] Even if the language "configured to" made a difference in *Typhoon Touch* (which it did not), Nazomi itself pressed a construction based on such language. In the district court, Nazomi argued: the "apparatus claims are directed to physical circuitry in CPU hardware that is *configured to* perform the recited functional language." A4728 (emphasis added). Having invited such a construction, Nazomi cannot now argue that the district court erred in adopting it. *Interactive Gift Express, Inc. v. Compuserve Inc.*, 256 F.3d 1323, 1345-48 (Fed. Cir. 2001) (holding that party may be judicially estopped from asserting inconsistent positions on claim construction and that appellate arguments that change position on claim construction are waived); *Key Pharms. v. Hercon Labs. Corp.*, 161 F.3d 709, 715 (Fed. Cir. 1998) ("The impropriety of asserting a position which the trial court adopts and then complaining about it on appeal should be obvious on its face, and litigants hardly need warning not to engage in such conduct.").

undisputed facts here. In *Fantasy Sports*, the patentee claimed a computerized fantasy football game. One limitation required the awarding of "bonus points" – a term that was construed to mean "points that are awarded 'in addition to the normal points for a scoring play.'" *Fantasy Sports Props., Inc. v. Sportsline.com, Inc.*, 287 F.3d 1108, 1112 (Fed. Cir. 2002). This Court held that the accused product infringed because it was "capable of" practicing the bonus point limitation.

Significantly, the Court explained that its conclusion was not based on the product being "capable of being altered in an infringing manner." *Id.* at 1117. Rather, the product was designed as "a modifiable software tool" that allowed users to set up "fantasy football leagues on customized internet web pages." *Id.* Thus, users of the product could access "options" that caused the claimed "bonus point" functionality to occur. *Id.* at 1118. Accessing this functionality required no alteration or modification of the accused product. Rather, the user only needed to activate "means that are *already present in the underlying software*" without having to "alter the code." *Id.* (emphasis in original). As explained *supra* and *infra*, the undisputed facts show that is not so with respect to the Slingbox or the My Book.

Nazomi also points to *Finjan*, arguing that this Court did not require the claimed functionality to be active or enabled and that inactive structures alone satisfied the claims. Nazomi Br. 34. But *Finjan* simply applied this Court's clear

rule to facts unlike those here. The claimed functionality there could be used merely by purchasing an activation key from the defendants that unlocked software already present in the accused device. Nothing in the accused device had to be altered, modified, or taken apart for it to infringe.

In *Finjan*, the accused product was a computer security device that provided customers with a menu of options: the device contained a number of locked modules where a customer could "purchase a separate key to activate each individual module." *Finjan*, 626 F.3d at 1202. Some of these modules, when unlocked by entry of the separately purchased activation key in accordance with the defendant's design, practiced the claimed functionality. But to use that functionality, the device did not need to be modified and no software needed to be installed.

This Court explained that "[t]here is no evidence that customers needed to modify the underlying code to unlock any software modules." *Id.* at 1205. Rather, "it [wa]s undisputed that software for performing the claimed functions existed in the products when sold – in the same way that an automobile engine for propulsion exists in a car even when the car is turned off." *Id.* Relying on the analysis in *Fantasy Sports*, the Court in *Finjan* highlighted that the product was designed so that a user could "utilize the function . . . without having to modify" the device. *Id.* (ellipses in original) (quoting *Fantasy Sports*, 287 F.3d at 1118). Moreover, on the

facts of *Finjan*, it was clear that defendants intended that users be able to buy the key to activate the function. *Id.* at 1202, 1205. In stark contrast (as explained *supra* and *infra*), the accused devices here are not capable of practicing the claimed functionality without being significantly modified, and end users are not intended to so modify them (and likely could not even if they tried). *See* pp. 18-22 *supra*; pp. 50-56 *infra*.

Nor did *Silicon Graphics* hold that dormant circuitry necessarily practices the functional language in apparatus claims. *Contra* Nazomi Br. 39-40. The claims in *Silicon Graphics* were directed at a computer graphics system, and a limitation in the asserted claims "require[d] 'a rasterization circuit coupled to the processor that rasterizes' primitives in a certain way and 'a frame buffer coupled to the rasterization circuit for storing' certain data." *Silicon Graphics*, 607 F.3d at 795. The defendant did not sell computer graphics systems; instead, it sold processors that could be incorporated into computer graphics systems. *Id.* at 788-89.

Thus, as the district court correctly recognized (and Nazomi essentially ignores), the defendant in *Silicon Graphics* was sued for indirect infringement. A10. The issue was whether the defendant's processor, when incorporated into a computer graphics system, practiced the "rasterization" and "frame buffer" limitation. *Silicon Graphics*, 607 F.3d at 795. It did. In that context, this Court

explained that the processor as designed by the defendant was "capable of" satisfying the claim limitation. Nothing in the accused processor needed to be modified; it just needed to be used with an operating system exactly as the defendant intended. *Id.* at 794 (holding that a product can infringe "so long as the product is designed 'in such a way as to enable a user of that [product] to utilize the function . . . without having to modify [the product]'" (alterations in original) (quoting *Fantasy Sports*, 287 F.3d at 1118)).

The same is true in *Intel*. *Intel Corp. v. ITC*, 946 F.2d 821 (Fed. Cir. 1991). There, the asserted patent disclosed "an integrated circuit read-only memory with a programmable selection means for selecting alternative addressing modes." *Id.* at 831. The accused memory chips directly infringed because the claims recited an apparatus that could be programmed to perform the claimed functionality. *Id.* at 832. This Court explained: "Because the language of claim 1 refers to 'programm*able* selection means' and states 'whereby *when* said alternate addressing mode is selected' (emphases added), the accused device, to be infringing, need only be capable of operating in the page mode." *Id.* (emphases in original). Thus, the accused product there could practice what the asserted patent claimed because the product was programmable. *Id.* As this Court has explained: "*Intel* therefore does not stand for the proposition, as argued by [the patentholder], that infringement may be based upon a finding that an accused product is merely

capable of being modified in a manner that infringes the claims of a patent." *Fantasy Sports*, 287 F.3d at 1117-1118; *accord High Tech Med. Instrumentation*, 49 F.3d at 1555-1556 ("All that was required by the limitation at issue in *Intel* was that the claimed invention, an integrated circuit memory device, was 'programmable' to operate in a certain manner. The accused device, although not specifically designed or sold to operate in that manner, could be programmed to do so.").

Finally, Nazomi's reliance (Nazomi Br. 26) on *Hewlett-Packard*'s statement that "apparatus claims cover what a device *is*, not what a device *does*" is misplaced. *Hewlett-Packard Co. v. Bausch & Lomb Inc.*, 909 F.2d 1464, 1468 (Fed. Cir. 1990) (emphasis in original). The Court made that statement in the context of an obviousness challenge, where the alleged infringer argued that the patentee must "show 'operational differences' of the claimed device over the prior art." *Id.* This Court rejected that proposition: "[a]n invention need not *operate* differently than the prior art to be patentable, but need only *be* different." *Id.* (emphasis in original). So while the statement on which Nazomi relies is true as far as it goes, *Hewlett-Packard* says nothing about the issue here: what is necessary for a device, as sold, to infringe an apparatus claim that recites functional limitations.

II. APPLYING THIS COURT'S PRECEDENT, THE DISTRICT COURT CORRECTLY HELD THAT THE SLINGBOX AND THE MY BOOK DO NOT INFRINGE THE ASSERTED CLAIMS

The district court correctly applied this Court's well-settled legal rule to the asserted claims and to the undisputed facts regarding the Slingbox and the My Book. Applying that rule, the district court construed the asserted claims "to cover a hardware apparatus configured to process Java bytecodes without modification by an end-user." A8. The court further held that neither the Slingbox nor the My Book infringe the asserted claims, because neither is capable of practicing the asserted claims without significant, unauthorized user modification – modification that might render the device inoperable.

A. Each Asserted Claim Recites Functional Limitations

Nazomi first challenges the district court's claim construction. Nazomi contends that the asserted claims are "structural rather than functional" and are not required to be "configured to utilize the claimed CPU in a certain way." Nazomi Br. 15. But Nazomi's claims do not recite just a generic structure, like a CPU. The asserted claims expressly require a CPU with structural elements, such as a hardware accelerator, that can process both register-based *and* stack-based instructions. A100-A101 (claims 48 and 74); A131-A132 (claims 1 and 5). Indeed, that was the claimed improvement over prior art. Nazomi cannot now disclaim that functionality as unnecessary to practice its asserted claims.

***The claims*.** As the district court concluded, each claim recites an apparatus that can practice the claimed functionality – not an apparatus that can be modified or programmed to practice the claims. A5-A7. Claims 48 and 74 of the '362 patent recite four structural elements, and each element is defined by specific functions it performs when stack-based instructions are processed. A100-A101 (col.10:57-11:6); A101 (col.12:29-47). Claim 48 states:

> 48. A central processing unit (CPU), capable of executing a plurality of instruction sets comprising:
>
> an execution unit and associated register file, the execution unit *to execute instructions of a plurality of instruction sets, including a stack-based* and a register-based instruction set;
>
> a mechanism to maintain at least some data for the plurality of instruction sets in the register file including *maintaining an operand stack for the stack-based instructions* in the register file and an indication of a depth of the operand stack;
>
> a *stack control mechanism* that includes at least one of an overflow and underflow mechanism, wherein at least some of the operands are moved between the register file and memory; and
>
> a mechanism *to generate an exception in respect of selected stack-based instructions*.

A100-A101 (col.10:57-11:6) (emphases added). Similarly, claim 74 recites structural elements with particular functionality, including "a decoding mechanism to decode instructions of a plurality of instruction sets including a -stack-based

[sic] instruction set and a register-based instruction set" and "an execution unit that processes the output of the decoding of the instruction of the stack-based instruction set." A101 (col.12:30-33, 42-43).

There is no language in these claims suggesting that they encompass (or even contemplate) subsequent user modification. *Silicon Graphics*, 607 F.3d at 794 (apparatus claim with functional language requires device to practice claim without unauthorized modification); *Typhoon Touch*, 659 F.3d at 1379-1380 (same). The claims do not recite a mechanism or execution unit that can be modified to perform the claimed functionality. Nor are any of the other structural elements in the claims drafted such that they need only be programmable or configurable to perform the claimed functionality. *Intel*, 946 F.2d at 832.[5]

The two asserted claims of the '436 patent also recite functionality that must be present in the claimed structures. A131-A132 (col.14:30-45; col.15:26-50). Claim 1 of the '436 patent recites a "hardware accelerator to process stack-based instructions." A131 (col.14:36). The claim expressly states that the "hardware

[5] Nazomi points to the fact that the preamble of claim 48 uses "capable of." Nazomi Br. 22. But as discussed above, under this Court's well-settled rule, a "capable of" limitation requires an apparatus to be capable of practicing the claimed functionality without unauthorized user modification. *See* pp. 31-42 *supra*. In any event, this Court has held that, in general, "the preamble does not limit the claims." *Allen Eng'g Corp. v. Bartell Indus.*, 299 F.3d 1336, 1346 (Fed. Cir. 2002).

accelerator" "generates," "computes," and "pushes." A131 (col.14:38-44). It does not state that it is programmable or configurable to perform that functionality. And claim 5 of the '436 patent includes, for example, "a bytecode buffer that *receives* stack-based instructions from the memory." A132 (col.15:41-42) (emphasis added). Nothing in this claim language suggests that the apparatus need be only modifiable or programmable to satisfy the claimed limitations.

The specification. The specifications further undermine Nazomi's argument that the claims do not require a device capable, as provided, of processing stack-based instructions in hardware. *Typhoon Touch*, 659 F.3d at 1382 (specification supporting "capable of" construction). The specifications highlight that a major disadvantage in the prior art was using software to translate Java bytecodes into register-based instructions. A96 (col.1:45-48); A125 (col.1:42-48). Prior art methods were slower; increased speeds required less energy efficiency or increased memory requirements. A96 (col.1:54, 61-67); A125 (col.1:55-67). The '436 and '362 patents purported to solve this problem: unlike the prior art, "at least part of the Java™ Virtual Machine is implemented in hardware as the Java™ hardware accelerator." A96 (col.2:15-17); A125 (col.2:6-9). Indeed, the specification teaches that "[t]he hardware Java™ accelerator can convert the stack-based Java™ bytecodes into a [sic] register-based native instructions on a CPU." A96 (col.2:23-25); A125 (col.2:19-22). And Figure 1 (A87; A106) shows that "the translation of

the Java™ bytecodes into native processor instructions is at least partially done in the hardware Java™ accelerator." A96 (col.2:65-67); A126 (col.3:51-53).

Attempting to minimize the importance of processing Java bytecodes, Nazomi criticizes the district court for narrowing the claims to a single purpose. Nazomi Br. 15-16, 23. But the district court did not narrow the claims solely to a hardware accelerator that processes Java bytecodes. The district court merely explained that the asserted claims require the capability of processing stack-based instructions in hardware, and it noted that the specifications "consistently describe the invention as a hardware apparatus used to process Java bytecodes." A5-A6. Given the extensive references to the processing of Java bytecodes throughout the intrinsic record, this was not error.

Nor does it help Nazomi that the specifications teach that one of the "'major advantages' of the claimed CPU is the ability 'to increase the speed in which the Java Virtual Machine operates, *and* allow existing native language legacy applications, software base, and development tools to be used.'" Nazomi Br. 24 (emphasis in original). All that means is that the claimed invention must have *both* a "Java™ hardware accelerator" capable of processing Java™ bytecodes into native instructions *and* the ability to bypass that hardware so register-based native instructions can be sent to the CPU. A97 (col.3:31-37); A126 (col.4:23-29). In other words, the claimed apparatus is not "[a] dedicated microprocessor in which

the Java™ bytecodes [are] the native instructions." A97 (col.3:35-37); A126 (col.4:27-28). Because the district court never construed the claimed invention in such a limited manner, no embodiment was "read out" by the district court's interpretation of the claims.

The prosecution history. The prosecution history further confirms that processing stack-based instructions in hardware was essential to the asserted claims and that a device that is not capable of processing stack-based instructions in hardware simply practices the prior art. The PTO repeatedly rejected each asserted claim. To distinguish the prior art, Nazomi relied on the invention's use of hardware to process stack-based instructions. Indeed, the examiner made a written record of a face-to-face meeting in which Nazomi "agreed" that the "novelty" of its application was that the "execution unit *is executing* both register-based instructions and stack-based instructions." A4218 (emphasis added). Moreover, Nazomi consistently argued in written submissions that its claims were patentable because a hardware accelerator processed stack-based instructions. *See, e.g.*, A4313 (the claim "describes a system with a hardware accelerator *generating* new JAVA PC due to a JSR or a JSR-W bytecode *computing* the return JAVA PC and *pushing* the return JAVA PC on the operand stack") (emphasis in original); A4418-A4420 (same); A4079 ("[T]his claim includes the limitation of 'generating an exception in respect of a selected sta[ck-]based instruction while in the

accelerator mode', which limitation is not taught or suggested by" the prior art.). Having repeatedly argued during prosecution that the functions recited in the claims distinguished the invention from the prior art, Nazomi cannot now contend that those same functions do not matter. *Typhoon Touch*, 659 F.3d at 1381 ("The patentee is bound by representations made and actions that were taken in order to obtain the patent.").

Finally, Nazomi is wrong in arguing that the district court incorrectly construed the claims in light of the accused Slingbox and My Book hard drive. Nazomi Br. 23 n.4. The district court's claim construction nowhere focused on the accused Sling Media or Western Digital devices. A4-A8. And to the extent Nazomi's argument is directed to what satisfies this Court's "capable of" requirement, that is an infringement analysis under this Court's precedent. *Fantasy Sports*, 287 F.3d at 1118 (examining whether accused product was "capable of" practicing the claims in the infringement analysis).[6]

[6] The district court also properly performed a preliminary claim construction analysis of only the terms relevant to Sling Media's and Western Digital's summary judgment motions. This Court has held that, although claim construction is independent of the accused device, "it is convenient for the court to concentrate on those aspects of the claim whose relation to the accused device is in dispute." *Pall Corp. v. Hemasure Inc.*, 181 F. 3d 1305, 1308 (Fed. Cir. 1999).

B. The Sling Media And Western Digital Accused Devices Cannot Practice The Asserted Claims Without Significant, Unauthorized User Modification

Nazomi also challenges the district court's infringement analysis, but the district court correctly applied this Court's precedent. The Slingbox and the My Book are capable of practicing only the prior art that Nazomi distinguished from its invention during its prosecution of its patents.[7]

1. The Slingbox

The district court correctly held that the Slingbox is not "capable of" executing stack-based instructions. A9-A10. Nothing in the Slingbox can process stack-based instructions in a hardware accelerator as Nazomi's claims require. Nazomi does not dispute that the Jazelle circuitry is unusable in the Slingbox. Indeed, Nazomi concedes that Sling Media "did not choose to enable" Jazelle in the Slingbox. Nazomi Br. 35. Nor does Nazomi dispute that Jazelle circuitry can only be used with the proprietary JTEK software, which the Slingbox does not have and Sling Media does not provide. Without JTEK, the Jazelle hardware

[7] Although Nazomi relies on some indirect infringement cases (such as *Silicon Graphics*), Nazomi has not raised on appeal (and thus has waived) any indirect infringement claim. Indeed, Nazomi did not oppose summary judgment as to indirect infringement. A8 n.1 (granting summary judgment on indirect infringement because Nazomi "did not respond to moving defendants' arguments").

within the ARM processor does not function and is not capable of processing anything – let alone Java bytecodes. A3614 ¶16; A2722-A2724 ¶¶5-8.

In fact, the Slingbox does not use any Java programming and is not capable of processing stack-based instructions at all. A3886 ¶3. Sling Media's device neither uses nor stores any stack-based instructions, and has no software that would allow a user to download or save stack-based instructions. A9; A3886 ¶3. Indeed, the Slingbox does not even have a JVM. A4981-A4982.

Under this Court's clear rule (*see* Part I *supra*), there can be no infringement of the asserted claims. Sling Media's device is not capable of practicing the asserted claims. *Typhoon Touch*, 659 F.3d at 1380. The Slingbox is not "designed 'in such a way as to enable a user of that [product] to utilize the function . . . without having to modify [the product].'" *Silicon Graphics*, 607 F.3d at 794 (alterations in original) (quoting *Fantasy Sports*, 287 F.3d at 1118).

Nazomi suggests that summary judgment of non-infringement must be reversed simply because its expert declaration from Dr. Babb went unrebutted. Nazomi Br. 48. But there is no such rule. It depends on what the declaration says. Here, Dr. Babb's declaration addressed only the *presence* of "physical hardware elements" (such as Jazelle) in the ARM926EJ-S processor core. The presence of that hardware was undisputed. But Dr. Babb never analyzed what mattered under this Court's law: whether that hardware in the Slingbox was capable of practicing

the claimed functionality without unauthorized user modification. A6528-A6533 at A6530-A6531 ¶¶12-13. In other words, to be "capable of" requires, at a minimum, some sort of intended availability to the user – something Dr. Babb never addressed. Dr. Babb's opinion thus could not defeat summary judgment because the mere "presence of physical hardware elements found in those CPUs" (A6530 ¶13) does not demonstrate infringement. *Typhoon Touch*, 659 F.3d at 1380; *Fantasy Sports*, 287 F.3d at 1118.

Nazomi also ignores that the Slingbox would have to be significantly modified to be capable of practicing the asserted claims. A9; *Finjan*, 626 F.3d at 1205 (infringement requires that a user could use the functionality "without having to modify" the device). As explained above (Part I *supra*), a device infringes if it practices the claims "as designed, sold, and intended for use." *High Tech Med. Instrumentation*, 49 F.3d at 1555. The possibility of modification provides a basis for infringement only "if a device is designed to be altered or assembled before operation." *Id.* at 1556.

Here, Sling Media never intended the Slingbox to be modified by customers. A3886-A3887 ¶6. Unlike the "customizable" options in *Fantasy Sports* and the simple key code in *Finjan* that unlocked already present functionality, the Slingbox cannot be made capable of processing stack-based instructions by entering a key or code, selecting an option, or turning on a switch. *Fantasy Sports*, 287 F.3d at

1118; *Finjan*, 626 F.3d at 1205. And unlike the products in *Fantasy Sports* and *Finjan*, the Slingbox does not have software already installed or configured that enables the device to infringe. *Fantasy Sports*, 287 F.3d at 1118 ("the user is only activating means that are already present in the underlying software") (emphasis omitted); *Finjan*, 626 F.3d at 1205 ("it is undisputed that software for performing the claimed functions existed in the products when sold").

Instead, for a customer to place Java bytecodes on the Slingbox, a customer would need to "hack" the product – without any authorization from Sling Media – and manually add those instructions onto the device. A3886 ¶4. Even if the Slingbox could be hacked in this manner (and Sling Media is aware of no instances where such hacking has occurred), the Jazelle circuitry still would not function because Sling Media does not license (and does not provide) the JTEK software. A3890 ¶¶2-5. That software would have to be somehow loaded onto the device. The most likely result from such an endeavor would be a broken, non-functional, and useless Slingbox. A3886-A3887 at ¶¶2-6. Thus, any attempt to modify the Slingbox would not "serve any functional purpose." *High Tech Med. Instrumentation*, 49 F.3d at 1556.

2. The My Book

The district court also correctly held that Western Digital's My Book is not capable of processing Java bytecodes in hardware. A8-A10. The My Book

processes no stack-based instructions in hardware. It uses only software (specifically, Just-in-Time conversion) to convert Java bytecodes. A2745 ¶3; A3723; A3724; A3730. Just like Sling Media's product, Western Digital has never licensed ARM's JTEK software. A3724; A3735-A3736; A3743. Thus, the Jazelle hardware in the My Book as sold is not capable of doing anything. A2745-A2746 ¶¶3-11; A3724; A3735-A3736; A3743.

Indeed, the record here is undisputed. Nazomi sought extensive discovery, and took three Rule 30(b)(6) depositions. Those depositions confirmed three unassailable facts. *First*, the accused My Book does not do any stack-based processing in hardware. Western Digital's 30(b)(6) witness testified at length on this point. Western Digital uses only "JIT" conversion, which uses software to convert Java bytecodes. A3723 ("[T]he JVM provided by Sun did not use Jazelle as acceleration technology. It actually used the HotSpot or JIT technology for the acceleration."); A3724; A3730 ("JIT is an independent kind of technology that does not use Jazelle . . .").

Second, the accused My Book as sold to customers does not enable or use Jazelle. "For products that were shipped to customers, Western Digital never included a JVM that utilized Jazelle technology." A3735; *see* A3724; A3735-A3736 ("the shipping product clearly would not have included a patch to Jazelle since it had no usage of Jazelle"); A3737; A3743.

Third, the accused My Book cannot be reasonably modified to use Jazelle. To enable the Jazelle circuitry in the My Book, an end user would have to open the plastic casing of the product, take the product apart, connect the internal circuitry to a computer, and "hack" the system and replace substantial portions of the software and/or operating system. A3745; A3750; A3745-A3756.

This would be no simple task. The end-user first would have to modify JVM software that is hidden from the user. "[T]hat Java Virtual Machine is located on the hidden portions of the disk drive." A3750. Such a modification is not authorized or approved by Western Digital and "voids any warranty on the product." A3750.

Even if the end-user were able to locate the "hidden" software, the user would still need to know exactly which files to try to replace:

> To modify the JVM in My Book World Edition, the user would have to either replace all of the software which processes Java or the user would have to know exactly how to replace the specific portions of that software.

A3755. To Western Digital's knowledge, no one has ever modified the software to be able to use Jazelle: "Q. Do you know if anyone has ever updated the JVM to use the Jazelle hardware? A. The JVM has not been updated to use the Jazelle hardware." A3741; A3745-A3756.

Finally, even if someone were able to modify the software to try to use Jazelle, there is no evidence that the Jazelle circuitry within the Western Digital

product functions at all. Western Digital could never get the Jazelle hardware to function successfully. A3730 ("[W]e never had the JVM in a state using Jazelle technology that was fully usable."). Indeed, the district court specifically noted: Western Digital had "produced uncontroverted testimony that a hardware-based JVM 'was not stable enough' for use on the MyBook, and that it elected to employ a software-based solution instead." A9.

In short, the My Book processes stack-based instructions, using software, in the same manner as was done long before Nazomi applied for the asserted patents. A9. Accordingly, as with the Slingbox, the My Book is not capable of processing Java bytecodes in hardware: Jazelle is not enabled in the accused Western Digital product and the Western Digital My Book is not reasonably capable of being modified to enable Jazelle.

C. Nazomi Incorrectly Focuses On The Processor Core Rather Than The Accused Devices

Nazomi contends that the district court erred by analyzing the accused products as a whole, rather than focusing solely on the CPU or the circuitry within the CPU. Nazomi Br. 32-34, 42. Under this Court's case law, that is not the relevant inquiry. But in any event, even if one were to ignore the rest of the accused device, and look only at the devices' respective CPUs (or the processor cores inside the CPUs), those components are not capable of practicing the claimed invention either.

As an initial matter, this Court consistently has examined whether a device, not a component (or circuitry within a component), practices the claim. *Finjan*, 626 F.3d at 1203; *Typhoon Touch*, 659 F.3d at 1380; *Fantasy Sports*, 287 F.3d at 1118; *High Tech Med. Instrumentation*, 49 F.3d at 1555. Nazomi cites no case where this Court has found infringement for a device as sold on the ground that a component of the device could practice the claims if configured differently, if incorporated into a different product, or if used in an entirely different manner.

Nazomi's reliance on *SunTiger* is misplaced. Nazomi Br. 36. That decision merely stands for the uncontested and well-settled principle that an alleged infringer "cannot avoid infringement merely by adding elements if each element recited in the claims is found in the accused device." *SunTiger, Inc. v. Scientific Research Funding Grp.*, 189 F.3d 1327, 1336 (Fed. Cir. 1999) (internal quotation marks omitted). But that principle does not help Nazomi here: no aspect of the accused devices practices all the elements of the asserted claims.

The asserted patent in *SunTiger* was "directed to optical lenses for sunglasses that screen out certain wavelengths of visible light that may cause harm to human eyes." *Id.* at 1329. A transmission limitation required the lens to incorporate orange dye that allowed the transmission of 90% of certain visible sunlight while "block[ing] more than 99%" of other types of sunlight. *Id.* at 1331. The accused lens incorporated orange dye "that ha[d] the same transmission

characteristics as set forth" by the claim. *Id.* But the "accused lens also incorporate[d] a gray gradient surface coating that reduce[d] the amount of visible light transmitted by the accused lens." *Id.* Thus, when the accused product was tested, "no part of the accused lens met the transmission limitations of [the claim] except for the 'right bottom' portion of the accused lens where the gray gradient coating was lightest and allowed for the transmission of the most visible light." *Id.*

This Court held that the district court should not have granted summary judgment of noninfringement for two reasons – neither of which applies here. First, even though the gray gradient surface coating prevented *additional* light from being transmitted, it did not alter the inherent features of the orange dye which affected the transmission of light. *Id.* at 1336. In other words, regardless of the effects of the gray gradient surface coating, the orange dye was reducing the transmission of light in a manner that practiced the claim. Second, the "right bottom" portion of the accused lens – where the gray gradient surface coating was the lightest – practiced the transmission limitation and was unaffected by the additional coating. *Id.* Thus, even though the entire lens might not practice the transmission limitation, the Court observed that "[i]f a claim reads merely on a part of an accused device, that is enough for infringement." *Id.*

Here, even if one looked at the CPU within each device in isolation, the processor cores within those CPUs are not capable of practicing the claimed

functionality. The CPUs that Sling Media and Western Digital respectively purchase are not capable of infringing; Sling Media and Western Digital do nothing to them to make them capable of infringing; and the CPUs are not capable of infringing when imbedded in the Slingbox or the My Book. Jazelle is inactive at all times. And there is no way to activate it within the accused devices. *See* pp. 19-20 *supra* (Slingbox); pp. 21-22 *supra* (My Book). Thus, unlike *SunTiger*, Nazomi's claims do not read on any part of the accused products because neither the Slingbox nor the My Book, nor their respective CPUs, can process Java bytecodes in a hardware accelerator. Indeed, the Slingbox is not capable of processing Java bytecodes at all.

Nor is Nazomi aided by *Revolution Eyewear, Inc. v. Aspex Eyewear, Inc.*, 563 F.3d 1358 (Fed. Cir. 2009). Nazomi contends that *Revolution Eyewear* demonstrates that the district court should have focused on whether certain CPU circuitry could have been activated and used, without regard to whether any modifications might be necessary to use that circuitry. Nazomi Br. 45-48. But in *Revolution Eyewear* the accused device infringed without any modification of the product. Indeed, the Court noted that the defendant "appear[ed] to concede that its product included all the limitations of" the asserted claim. *Revolution Eyewear*, 563 F.3d at 1369.

In *Revolution Eyewear*, the relevant claim required a primary eyeglass frame with "magnetic members capable of engaging second magnetic members of an auxiliary spectacle frame so that lenses of an auxiliary spectacle frame are located in front of said primary lenses." *Id.* at 1363. The accused eyeglasses incorporated such magnetic members, but the defendant made no auxiliary frames that could attach to the magnetic members. *Id.* at 1369. The patentee demonstrated that the accused devices nevertheless had the claimed functionality by creating three specially-made auxiliary frames and showing that the "magnetic members" of the accused primary frame could engage the auxiliary frames. *Id.*

The Court concluded that the accused frame infringed the claim. The Court explained that the claim was directed only to the primary frame; it did not claim an auxiliary frame. *Id.* at 1370. The accused device was a primary frame "capable of being used in" a way that would infringe the asserted claim, and did "not need to be altered in any way for it to be 'capable of engaging'" auxiliary frames. *Id.*

The undisputed facts here are different. The Slingbox and the My Book are not "capable of being used in a way" that would infringe Nazomi's claims. Quite the contrary: both devices would need to be "hacked," software would need to be installed, and even then the Jazelle circuitry would remain dormant and inactivated. A3886-A3887; A3890; A3729; A3745; A3750.

D. The District Court Did Not Impose An Intent Requirement

Nazomi asserts that the district court incorrectly required intent to find direct infringement. Nazomi Br. 37-39. But the district court imposed no intent requirement.

Under this Court's precedent, when a device as sold does not infringe, this Court has gone on to examine whether the device was designed with subsequent user modification in mind. If so, the maker might be liable for infringement even though the device as sold was not capable of infringement. *See* pp. 31-33 *supra.* For example, in *High Tech Medical Instrumentation*, this Court examined the device "as designed, sold, and intended for use," and held that it did "not infringe simply because it is possible to alter it in a way that would satisfy all the limitations of a patent claim." *High Tech Med. Instrumentation*, 49 F.3d at 1555. The Court observed, however, that "if a device is designed to be altered or assembled before operation, the manufacturer may be held liable for infringement if the device, as altered or assembled, infringes a valid patent." *Id.* at 1556.

Here, the district court correctly applied this Court's precedent. Because the Slingbox and the My Book do not infringe as sold, the district court considered whether Sling Media and Western Digital nevertheless might be liable due to the possibility of subsequent user authorization. The district court concluded that it was undisputed that neither Sling Media nor Western Digital had designed or

intended its accused product to be modified so as "to be configured" for the processing of Java bytecodes. A8-A9. Nazomi does not challenge that determination on appeal.[8]

CONCLUSION

This Court's precedent dictates that for an accused device to infringe a claim with functional language, the device as provided must be capable of performing the claimed functionality, without unauthorized user modifications. The district court correctly applied that rule to the asserted claims and the undisputed facts. Each asserted claim recites structures to process stack-based instructions in a hardware accelerator. Yet neither the Slingbox nor the My Book process any stack-based instructions in a hardware accelerator, nor are the devices intended to be modified to do so. The accused devices therefore cannot infringe the asserted claims, and the district court's judgment of non-infringement should be affirmed.

[8] Nazomi cites *Intel* as rejecting a requirement that an accused device's infringing modification must have been intended by the alleged infringer. Nazomi Br. 38. But the *Intel* Court rejected a different notion. As discussed above (*see* p. 41 *supra*), *Intel* involved a claim that referred to "'programm*able* selection means' and state[d] 'whereby *when* said alternate addressing mode is selected.'" *Intel*, 946 F.2d at 832. In deciding whether that claim had been directly infringed, the Court held inapposite "induced and contributory infringement" precedent and rejected the defendant's argument that "an alleged infringer must intend its [unmodified] parts to be used in an infringing fashion." *Id.* That says nothing about the issue here: whether a defendant can be held liable for direct infringement of claims that do *not* encompass subsequent user modifications, where the defendant never contemplated or intended infringing user modifications.

Dated: June 14, 2013

Respectfully submitted,

By: /s/ Deanne E. Maynard

DEANNE E. MAYNARD
BRIAN R. MATSUI
MORRISON & FOERSTER LLP
2000 Pennsylvania Avenue, N.W.
Washington, D.C. 20006-1888
Telephone: 202.887.8740

RACHEL KREVANS
SCOTT C. MOORE
ESTHER KIM
MORRISON & FOERSTER LLP
425 Market Street
San Francisco, CA 94105-2482

Counsel for Defendant-Appellee Sling Media, Inc.

Dated: June 14, 2013

Respectfully submitted,

By: /s/ Kevin P. Anderson

KEVIN P. ANDERSON
KARIN A. HESSLER
ROBERT J. SCHEFFEL
WILEY REIN LLP
1776 K Street, N.W.
Washington, DC 20006-2304

Counsel for Defendants-Appellees Western Digital Corporation and Western Digital Technologies, Inc.

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the foregoing was filed with the Clerk of the United States Court of Appeals for the Federal Circuit by CM/ECF on June 14, 2013. The undersigned further certifies that counsel of record are registered CM/ECF users and will be served by the CM/ECF system.

Dated: June 14, 2013

/s/ Deanne E. Maynard
Deanne E. Maynard

CERTIFICATE OF COMPLIANCE WITH RULE 32(a)

This Response Brief for Defendant-Appellee Sling Media, Inc. and Defendants-Appellees Western Digital Corporation and Western Digital Technologies, Inc. complies with the type-volume limitation of Rule 32(a)(7)(B)(i) of the Federal Rules of Appellate Procedure because it contains 13,299 words and is therefore below the limit of 14,000 words.

Dated: June 14, 2013

/s/ Deanne E. Maynard
Deanne E. Maynard

dc-713834